On the Design of Visual Behaviors for Autonomous Systems

José Santos-Victor, Alexandre Bernardino, César Silva

Instituto Superior Técnico/Instituto de Sistemas e Robótica
IST - Torre Norte
Av. Rovisco Pais, 1
1096 LISBOA CODEX, Portugal
e-mail: {jasv,ajmb,cesar}@isr.ist.utl.pt

Abstract – We describe a set of visual behaviors developed over the years, in the general context of robot navigation. The first set of behaviors solve three basic problems in vehicle navigation: obstacle avoidance, docking to a surface and moving along corridors or walls. Another set of visual behaviors were developed for the active stereo head Medusa, with the purpose of tracking objects with arbitrary shape or motion. Finally, we present an approach for egomotion estimation assuming arbitrary motion for the camera. All behaviors use purposive visual inputs, based on similar image measurements (image flow) but computed on different regions of the visual field. Goals are attained without camera calibration or environment reconstruction.

I. INTRODUCTION

Many animals perform numerous and repetitive tasks without the apparent consciousness of the actions involved. The mental processes involved in directing human locomotion look, at least in adult humans, surprisingly simple and do not seem to require a conscious activation of different processes or the explicit computation of environmental measures. All these processes seem to be happening "automatically" and they do in fact occur in parallel with other mental processes. In principle they cannot be considered "reflexes" (at least in the sense physiologists define them) because the same sensory input can cause rather different motor responses according to what the person is doing and the motor response can be voluntarily suppressed. However it is fair to say that the above mentioned processes seem to be running in our brain without a constant conscious supervision. In humans these behaviors are certainly the result of a developmental process during which a mapping between sensing and acting is built, giving rise, for example, to a sensory-motor representation of locomotion. Once these representation is learned, usually with trial-and-error procedures, the emerging behaviors become part of the daily routine.

What is largely learned in humans often comes as a reflex in so-called lower level animals which cannot afford a long period of learning before becoming "autonomous". Some of the approaches presented in this paper have been, in fact, inspired by insect behavior and is aimed at building a library of embedded visually-guided behaviors coping with the most common situations encountered during autonomous navigation.

From the perceptual point of view, the approach has its roots in the paradigm of active perception [3, 4, 2] in the sense that the behaviors are based on active exploration of the environment and take advantage of both the structure of the environment and the purpose of the task to be solved [1].

One of the most fundamental assumptions used throughout the paper is the continuous use of visual information *during* the action [6, 10]. The second is the possibility of designing a set of closed-loop visually guided behaviors whose goal is solely that of controlling direction of heading and forward velocity of the navigating actor (the goal of perception is to act), or controlling the degrees of freedom of a binocular head. The solutions presented do not, of course, solve the problem of system autonomy. Our goal, on the contrary has been to implement a set of behaviors which can economically solve a set of low-level (yet complex) problems, freeing the overall system (yet to be implemented) from some of the routine tasks encountered in autonomous systems.

II. APPROACH

A robot moving in a indoor environment safely requires, as basic skills, the ability to detect and avoid obstacles, to navigate along walls and corridors, to enter doors and approach surfaces with a given orientation (e.g. when entering an elevator or docking in a given place to collect or deliver materials), determine its own motion parameters and to track moving objects in the scene. The behaviors described here, even if not integrated yet, do in fact cover almost all of these requirements and they do it by using similar visual measures (which is good from the economical point of view) purposively used to control the motion variables specific to each behavior.

An important aspect is the way we use the optical flow to pursue our goals. Estimating the full optical flow field is an underconstrained problem due to the *aperture problem*, since with local image measurements we can only determine the flow component along the image gradient direction - the normal flow. In our approach, we exploit the particular nature of each behavior to constrain the required optical flow computations.

In Section III, we present the various behaviors related to navigation. The centering-reflex, observed for freely flying honeybees, is used to drive a robot along corridor or wall like environments, while controlling the forward speed and avoiding lateral obstacles.

Then we describe an obstacle avoidance behavior

based on the inverse perspective mapping and a set of docking behaviors to control the robot to a certain point in the scene, aligning itself perpendicularly to the docking surface and controlling the velocity.

In Section IV, we describe an approach for binocular tracking based on log-polar images that combines vergence and pursuit behaviors. Binocular disparity and motion cues are used to control the 4 degrees of freedom of our robot head.

Finally, in Section V we illustrate how the normal flow can be used to estimate the robot ego-motion which can be integrated in a closed loop control strategy. We rely exclusively on the information of the normal flow to recover the robot linear and angular velocity.

Therefore, by considering the specificities of each of the behaviors, we can tailor the visual processing required to achieve the control goals, and therefore overcome some of the limitations related to the *aperture problem*. Additionally different attentional areas are used for the various systems (e.g foveal versus peripheral vision) or in some cases attention is directly embedded on the sensor geometry.

III. NAVIGATION

We have designed of a set visual behaviors to solve some relevant problems in autonomous navigation: navigating along corridors or walls, obstacle avoidance and docking to a surface in the environment.

III.1. Centering Behavior

The first visually guided behavior is the *centering reflex*, described in [19] to explain the behavior of honeybees flying within two parallel "walls". The qualitative visual measure used is the difference between the image velocities computed over a lateral portion of the left and the right visual fields, as described in [11].

One of the major driving hypothesis is the use of qualitative depth measurements. Additionally, the goal of the visuo-motor controller is limited to the "reflexive" level of a navigation architecture acting at short-range. In spite of these limitations, we will demonstrate a variety of navigation capabilities which are not restricted to obstacle avoidance or to the "centering reflex". For example it has been shown the possibility of overcoming unilateral or bilateral absence of flow information, caused by absence of texture, or by localized changes in environmental structure (e.g. an open door along a corridor). An important remark is that the system does not critically rely on the accuracy of the optical flow computation, because the measurements are continuously used in closed loop.

Real-time Control

The overall structure of the robot control system involves two main loops. The *Navigation loop* controls the robot rotation speed in order to balance left and right optical flows, hence maintaining the robot at similar distances from structures on the right or left sides. The difference between the left and right flow vectors (along the horizontal direction) is used to control the robot heading. The *Velocity loop* controls the robot forward speed by accelerating/decelerating as a function of the amplitude of the lateral flow fields. The robot accelerates if the lateral flow is small (meaning that the walls are far away), and slows down whenever the flow becomes larger (meaning that it is navigating in a narrow environment). The mean flow vector on each side of the robot gives a qualitative measurement of depth.

Additionally, a ***sustaining mechanism*** is implicitly embodied in the control loops to avoid erratic behaviors of the robot, as a consequence of localized (in space and time) absence of flow information. It allows the use of the robot in environments far more complex than corridors, or when the "walls" are not uniformly covered with texture.

To overcome these problems, we have introduced in the control strategy a mechanism able to cope with unilateral lack of flow information. Whenever it occurs, the control system uses a reference flow that should be sustained on the "seeing" camera (i.e. the camera still measuring reliable flow). Hence, the robot will follow the ipsilateral wall at a fixed distance. This simple qualitative mechanism extends the performance of the system to a much wider range of environmental situations.

Results

In all the tests, the robot trajectory was recorded from odometric data during real-time experiments. Figure 1, shows the robot trajectories superimposed on the experimental setup, for various types of environment.

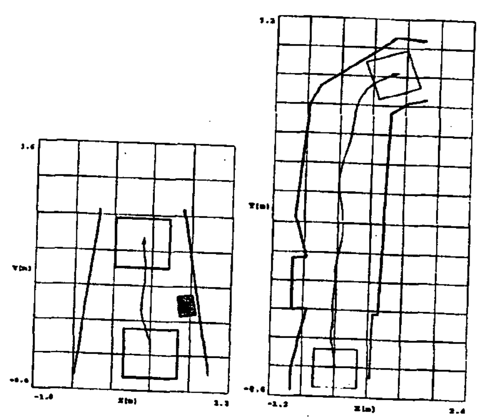

Fig. 1.: Results obtained in the closed loop operation for different scene layouts.

To test the velocity control, we considered the funneled corridor example, with varying width. As the corridor becomes narrower, the average flow increases and the velocity control mode forces the robot speed to decrease. Using the velocity control, the final turn of the corridor experiment is done at a reduced speed enabling the robot to make a softer, safer maneuver.

With the introduction of the sustained behavior, the robot is able to navigate in a much wider set of environments. In fact, only one textured wall is needed for the navigation strategy to work.

III.2. OBSTACLE DETECTION

The navigation system described before is unable to detect obstacles located ahead of the robot. Here, we describe a system for obstacle detection which uses a similar input to avoid obstacles. It exploits the geometric properties of the vehicle-camera-scene arrangement, and does not depend on the knowledge of the camera parameters or vehicle motion, as described in detail in [12].

The basic assumption is that the robot is moving on a ground floor (as it was the case of the centering behavior presented before) and any object not lying on this plane is considered to be an obstacle. The method is based on the inverse projection of the flow vector field onto the ground plane, where the analysis of the flow pattern is much simplified.

As opposed to other systems, an important feature is that the knowledge of the vehicle motion is not required and under certain circumstances, the approach is independent of the camera intrinsic parameters.

Inverse Perspective Mapping

The main idea is the analysis of the particular structure of the optical flow field, when a robot with a camera is moving over a ground plane. In Figure 2, *Pc* is the image plane of a camera moving forward with pure translation,

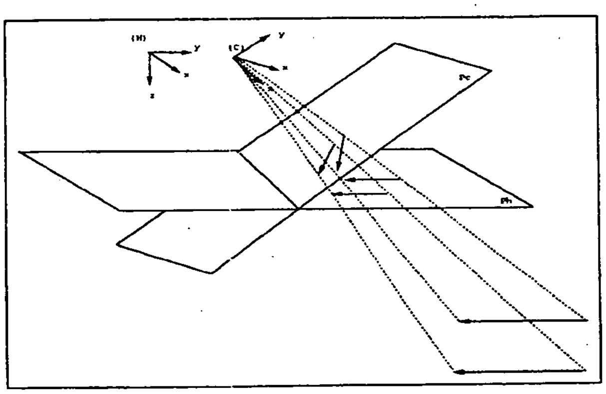

Fig. 2.: Inverse perspective mapping. The coordinate systems (C) and (H) share the same origin even though in the picture they have been designed separately for simplification. While on (C) the motion of the ground floor is perceived as a complex vectorial pattern, in (H) all the vectors have equal length and orientation under translational motion.

while facing a planar pavement. Even with this simple arrangement, the resulting flow pattern is complex. This is due to the perspective effects, and makes the problems of motion analysis or obstacle detection difficult.

The idea is to inverse project the flow captured on the image pane *Pc* onto the horizontal plane *Ph*, as suggested in [8]. In this coordinate plane, the flow pattern becomes much simpler as the distance to any point on the pavement is constant. All the vectors are equal, and points lying above or below the ground plane can be easily detected, as obstacles.

The method operates in two steps. Initially, the robot moves with pure translation and the projective transformation between the image and ground planes is estimated, without the need to calibrate the camera. During normal operation, the normal flow field is inverse projected onto the horizontal plane, where obstacles are easily detected.

The planarity assumption is used to approximate the flow field of the pavement to an affine mapping. The affine flow parameters are then used to estimate the projective transformation. The more salient features are the exclusive use of the normal flow information; and that knowledge about the vehicle velocity, or camera parameters is not required. Apart from the initialization phase, the system can deal both with rotational and translational motion, of the robot, and the method is appropriate for use in real-time.

Results

The method was tested extensively using synthetic and real image data, and in real time on a mobile robot. The camera was placed in the front part of the robot facing the ground plane with an angle of about 65 degrees, with no calibration. The robot speed was set to 10 cm/s. The running frequency of the vision loop is around 1Hz.

Figure 3 shows an example of the normal flow field measured during the robot motion, with an obstacle in the field of view. The rightmost image of Figure 3 shows the image regions where points lying outside the ground plane, have been detected. When the robot is undergoing pure linear motion, we have simply projected the inverse mapped flow in the y direction and subtracted the median flow. As the robot we used has a single forward component of the linear velocity, we have also estimated the rotation component and canceled this term from the overall inverse projected flow and then, the same detection process was applied. In both cases, similar results were obtained (see [12] for details).

Fig. 3.: Left: sample of the ground plane normal flow field, during the robot motion. Center: resulting inverse projected flow. Right: detected obstacle.

The resolution of the overall system determines the minimum obstacle size that can be detected, and strongly depends on the image resolution. If more computational power is available, the image resolution can

be increased, the same happening with the global system resolution.

III.3. DOCKING

In this section, we introduce vision-based docking strategies for indoor mobile robots, where the robot should approach a surface, along the surface normal with controlled forward speed, until it finally stops.

Two distinct situations for the docking problem are considered. In the *ego-docking*, the camera is mounted on board the vehicle, and the robot egomotion is controlled during a docking maneuver to a particular surface in the environment. In the second scenario, that we call *eco-docking*, the camera and computational resources are installed on a single external docking station with the ability to serve multiple robots. Both scenarios are illustrated in Figure 4.

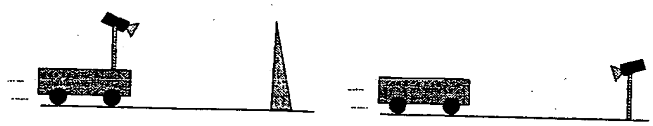

Fig. 4.: Left: In the *ego-docking*, a robot, equipped with a camera and computing resources, docks to a surface. Right: in the *eco-docking*, the camera is attached to a single docking station which may serve multiple robots.

From the perceptual point of view, both situations are quite similar since the important issue is the relative motion between the camera and the docking surface. However, in the *ego-docking*, the camera position with respect to the robot is fixed, whereas in the *eco-docking* it is changing continuously, thus posing new problems for the visuo-motor control loop. By proper formulation of the problem [13], exactly the same control architecture can be used in both cases.

The objective of the control system, both in the *ego-docking* and the *eco-docking* problems is twofold. The *Heading* control system aligns the camera axis and the docking surface normal, during the docking maneuver. The robot approaches the surfaces perpendicularly. Then, we use the *Time to crash* information to control the robot forward speed to slow the robot down when approaching a wall.

As before, this behavior uses the planar surface assumption to approximate the flow by an affine mapping [13]. The affine optical flow parameters are estimated from spatio-temporal image derivatives, and used directly to control the robot motion. One of these parameters is inversely proportional to the *time to collision*, and should be kept constant during the maneuver, slowing the robot when it approaches the goal. A second parameter vanishes when the proper orientation is reached, and can be regulated to zero to control the robot heading. The relation between the affine parameters and the robot linear and angular velocities is similar both for the *ego-docking* and *eco-docking*. Hence, apart from a sign inversion in the rotation controller, the same controller can be used for both situations.

Results

Figure 5 describes a typical ego-docking experiment, showing the robot trajectory (recovered from odometry) during the maneuver. Initially there is misalignment of about 45^0 between the robot heading and the direction perpendicular to the docking surface. During the maneuver the robot describes a smooth trajectory and aligns the camera axis with the direction normal to the surface, while controlling the forward speed. Figure 5

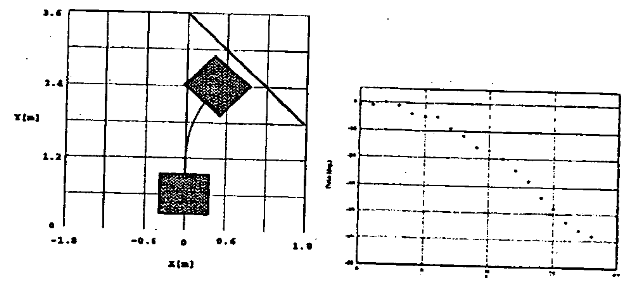

Fig. 5.: Robot trajectory during a real *ego-docking* maneuver and time evolution of the robot heading direction, in degrees.

shows also the time evolution of the robot heading direction during the maneuver.

Also in the eco-docking, the system has revealed a robust behavior and we have made several tests using different initial positions for the robot. We have used the same controller apart from a sign inversion in the rotation control law. Figure 6 shows the robot trajectory during a typical *eco-docking* experiment together with the evolution in time of the robot speed.

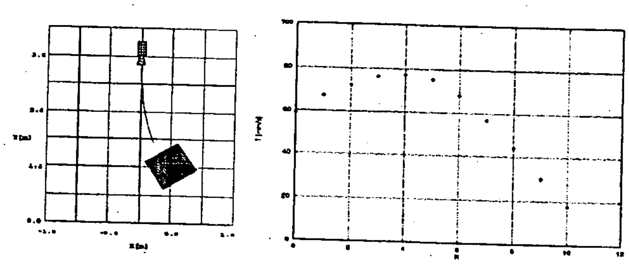

Fig. 6.: Robot trajectory during an *eco-docking* maneuver and time evolution of the robot forward speed in $[mm/s]$.

While the robot is far from the docking station, the speed control loop originates an increase of the robot velocity until a cruise speed. When the robot gets closer, the speed decreases.

The angular and position errors in the maneuvers are in the range of a few degrees (typically up to 5^0) in orientation and a few centimeters (typically up to $5cm$) in the distance to the docking surface. These errors are mainly due to the low resolution of the images we use, to the relatively low sampling frequency, and to mechanical problems in the platform when the speed is very small.

IV. BINOCULAR TRACKING

Many robot and computer vision problems can improve their performance by tracking objects in the visual field. The tracking system presented in this paper is implemented on an active vision stereo head, and follows some aspects motivated by the structure and functionality of biological visual systems, which show several advantages over other more straightforward approaches. In particular the use of binocularity, individual ocular movements, a space variant image representation, and visuo-motor strategies based on low-level visual cues in a closed loop control architecture, play a determinant role in the performance of the tracking system.

Binocular tracking systems have the ability to perceive target depth, which can be an important cue in many robotic tasks. Additionally, binocular fusion greatly simplifies figure-ground segmentation, which is a crucial step for most applications.

When using binocular active vision systems one has to address the problem of dealing with redundant perceptual information and motor degrees of freedom. We define two basic visuo-motor behaviors, Vergence and Pursuit, inspired in the most influent ocular movements in biological systems (vergence, saccadic and smooth pursuit). Each behavior only extracts the relevant information and controls the motions needed for its particular purpose. The vergence behavior controls the depth of the fixation point along the gaze direction while pursuit behavior controls the observation direction. Depth perception is attained through the extraction of disparity measures and directional information is obtained by a combination of target position and velocity in the image planes.

All perceptual measurements are made on space-variant resolution images. We use the log-polar mapping [15], which provides a geometry similar to the distribution of photo-receptors in the human retina, resulting in both perceptual and algorithmic advantages over the usual cartesian representation. Additionally, image size is reduced and the processing effort is concentrated in the center of the images resulting in faster algorithms and an implicit focus of attention in the center of the visual field. This last aspect is very important for tracking purposes because having higher resolution in the center of the images, where the target is expected to be, the areas belonging to the target are dominant in the performance of the algorithms, reducing the distracting influence of background elements in the periphery [5].

One of the main concerns of our approach is related with robustness and real-time functionality. The use of very fast perceptual techniques like correlation for disparity estimation and image flow for retinal target slip, integrated in a closed loop fashion, allow real-time and reliable performance despite the low-precision of the algorithms and system calibration errors. Moreover, the proposed visuo-motor strategies do not assume any previous knowledge about target shape or motion, providing high generality in the performance of the system.

The definition of several behaviors, competing or cooperating in the achievement of the same goal, usually brings coordination problems. In the present case, as vergence and pursuit behaviors are decoupled in both perceptual and motor aspects (acquire different stimuli and control different motions), they run in a parallel fashion with low internal dependency. However, when the vergence behavior can no longer guarantee correct binocular fusion on the target, motion estimation can become unreliable – in such conditions, vergence behavior inhibits the pursuit behavior. Despite this low internal dependency, vergence and pursuit are highly coupled in an external sense. First, pursuit brings the target to the central area of the images, enhancing the performance of the vergence behavior. Second, vergence provides binocular fusion on the target, enabling good figure-ground segmentation and motion estimation, required for proper pursuit. In this sense, the tracking problem is an example of how separable behaviors, tuned to different goals, cooperate towards a common purpose.

Results

The tracking system is implemented in the stereo head Medusa [14] and runs at 12.5 Hz (half video rate) without any specific processing hardware. All processing routines are implemented in a PENTIUM-200 computer, equiped with a PCI Frame Grabber, for image acquisition, and control boards, for motor control. Figure 7 shows a sequence of images obtained during a tracking experiment. In the beginning, fixation is stable in the door at the back of the room (images 1 and 2). Once the target (hand) gets close to the gaze direction, the system starts verging and tracking the target. Notice that the hand is always kept very close to the center of the images despite background motion, target rotation and scaling.

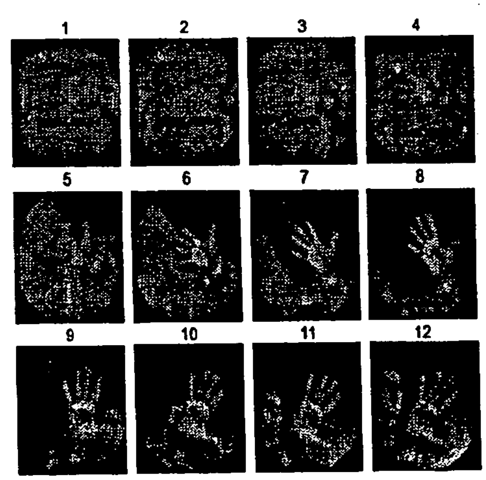

Fig. 7.: Hand tracking.

V. EGOMOTION ESTIMATION

In this section, we address the problem of egomotion estimation for a monocular moving observer. The prob-

lem consists in determining the 3D motion parameters, by observing an image sequence over time. This is a real need for many robotic applications where an autonomous system must be able to estimate and/or control its motion parameters before any higher level tasks can be addressed.

The first step to estimate egomotion is the computation of displacement between consecutive frames. Due to the known constraint of the *aperture problem*, the only image flow component that can be estimated based on local measurements is the normal flow.

The approach we follow is related to previous work [7, 16, 18], and the method consists in searching the image for particular geometric properties of the normal flow, tightly connected to the egomotion parameters. Hence, rather than considering the whole set of image flow data, we use only the image sites, that have special geometric properties, which convey relevant information about the observer motion. We split the search domain in several geometric figures, and estimate sets of motion parameters for each one of them.

At each image point, the normal flow provides a single constraint on the unknown components of the flow, and depends non-linearly on the translation and linearly on the camera rotation. Instead of using the flow all over the image, we select only the special image sites where the normal flow vectors do not depend on translation: they correspond to the set of normal flow vectors that are perpendicular to the lines radiating from the FOE (Focus of Expansion, the projection of the camera translation vector in the image plane). The idea is searching these special vectors to recover the FOE and the rotation (they depend linearly on the rotation). In general, the corresponding estimation methods involve computationally demanding search algorithms [18]. We develop a low complexity estimator by subdividing the search domain, according to some geometric constraints of the normal flow.

We use two types of normal flow vectors [16]: the *radial normal flow* (the set of normal flow vectors with radial directions) and the *circular normal flow* (the set of normal flow vectors with a direction perpendicular to radial lines). The translational component of the radial normal flow vanishes on the Γ-circle having the image center and the FOE as diametrical opposite points. The translational component of the circular normal flow is zero on the Ψ-line going through the FOE and the image center. On the other hand, the rotational component of the radial and circular normal flow is, respectively, sinusoidal in ψ and affine in r (where r and ψ are the polar coordinates of the image plane).

Two search algorithms can be defined based on these properties: **(1) The Ψ-line algorithm** searches a radial line passing through the image origin, such that the circular normal flow on this line is affine in r. Once the Ψ-line has been determined, this algorithm recovers uniquely two constraints on the rotation and the direction of the FOE. **(2) The Γ-circle algorithm** consists in searching a circle (having the origin and a point of the estimated Ψ-line as opposite points) such that the radial normal flow is sinusoidal in ψ. This algorithm solves the remaining egomotion parameters, namely the FOE and the individual rotation values.

The sequential method described here can be generalized for other search algorithms applied for different types of normal flow vectors [17].

These algorithms were tested in a series of experiments, using both synthetic and real image data. In Se-

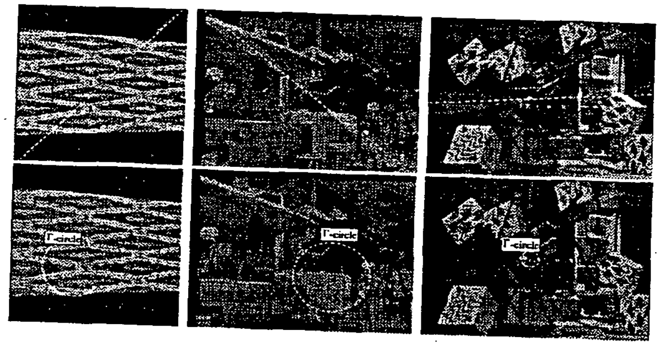

Fig. 8.: Results of the Ψ-line (top) and Γ-circle (bottom) algorithms using sequences 1, 2, 3.

quence 1 and 3, the camera rotates around the vertical axis, while translating. In Sequence 2, the camera undergoes a pure translational motion in a real cluttered scene. We have applied the Ψ-line and Γ-circle algorithms to all image sequences. The Ψ-lines and Γ-circles found are shown in Figure 8, and intersect close the true location of the FOE. We used a known robust estimator to estimate the rotational values from the corresponding observations: the *least median of squares* estimator [9]. The estimator is designed for a simple bidimensional estimation problem. See [18, 17] for more details about this estimation issue, namely a description of the rotational estimates (that are very close to the true values).

In summary, our approach depends solely on spatio-temporal image derivatives, and is based on the subdivision of the search domain in various subspaces, which depend on specific geometric constraints of the normal flow field. To decrease the sensitivity to measurement noise, we apply robust estimators on bidimensional estimation problems. Robustness can be improved if more subspaces are explored, but this choice depends on the required time of computation vs. the required accuracy of the estimates.

VI. CONCLUSIONS

We have presented various visual behaviors for mobile robots solving a set of relevant problems for autonomous systems. Navigation behaviors are used to control a robot when crossing a corridor-like environment, following walls, avoiding obstacles or approaching a given point in the scene with controlled orientation and speed. Vergence and pursuit behaviors are used to track moving objects in the scene, and egomotion behaviors permit to recover the vehicle motion parameters.

Apart from some specific differences, the approach adopted is based on a number of common principles.

One of the main aspects is the **purposive** definition of the sensory apparatus both at the geometric level (camera location and image geometry) and processing level (how to adjust the visual processing for each of the perceptual tasks). Even though the approach cannot be considered general, it solves with limited complexity relevant problems for an autonomous system.

Another issue worth mentioning is that **qualitative and direct** visual measures are used to achieve a reasonable autonomy with limited computational power. The approach is based on the **continuous use of visual measures**, providing a continuous stream of environmental information. Hence, these behaviors illustrate the possibility of implementing sensory-motor strategies where the need for a continuous motor control is not bounded by an "intermittent" flow of sensory information.

Two factors characterize the different behaviors: the part of the visual field where the attention is focused on and the fact that the control law adopted reflects the direct link between visual information and vehicle motion. An explicit use of such attentional mechanisms is done for the binocular tracking behavior, where a foveated sensor is used for the image representation.

The fact that the "appropriate action" is totally embedded inside each single behavior, is a very powerful way of breaking a complex problem into simpler, tractable ones. The complexity of the perceptual processes tuned to each specific behavior and the need for "general purpose" perceptual processes becomes less crucial. We strongly believe that in the long run, the economical advantage of this approach will become evident for artificial systems as it is already evident for natural living systems, and thus contribute to the design of truly autonomous systems.

VII. ACKNOWLEDGMENT

This work was partially supported by projects PRAXIS /3/3.1/TPR/23/94, JNICT-PBIC/TPR/2550/95 and EC ESPRIT/LTR NARVAL.

VIII. REFERENCES

[1] J. Aloimonos. Purposive and qualitative active vision. In *Proc. ECCV90 - European Conference of Computer Vision*, Antibes, France, April 1990.

[2] Y. Aloimonos, I. Weiss, and A. Banddophaday. Active vision. *Int. Journal of Computer Vision*, 1(4):333–356, January 1988.

[3] R. Bajcsy and C. Tsikos. Perception via manipulation. In *Proc. of the Int. Symp. & Exposition on Robots*, pages 237–244, Sydney, Australia, November 6-10 1988.

[4] D.H. Ballard. Animate vision. *Artificial Intelligence*, 48:57–86, 1991.

[5] A. Bernardino and J. Santos-Victor. Vergence control for robotic heads using log-polar images. In *Proc. of the 1996 IEEE/RJS International Conference on Intelligent Robots and Systems*, pages 1264–1271, Osaka, Japan, November 1996. IEEE Computer Society Press.

[6] C. Fermüller. Navigational preliminaries. In Y. Aloimonos, editor, *Active Perception*. Lawrence Erlbaum Associates, 1993.

[7] C. Fermüller. Qualitative egomotion. *IJCV*, 15(1/2):7–29, June 1995.

[8] H. Mallot, H. Bulthoff, J. Little, and S. Bohrer. Inverse perspective mapping simplifies opticla flow computation and obstacle detection. *Biological Cybernetics*, 64:177–185, 1991.

[9] P.J. Rousseeuw and A.M. Leroy. *Robust Regression & Outlier Detection*. John Wiley & Sons, Inc, 1987.

[10] G. Sandini, F.Gandolfo, E.Grosso, and M.Tistarelli. Vision during action. In Y. Aloimonos, editor, *Active Perception*. Lawrence Erlbaum Associates, 1993.

[11] J. Santos-Victor, G.Sandini, F.Curotto, and S.Garibaldi. Divergent stereo in autonomous navigation : From bees to robots. *Int. Journal of Computer Vision, Special Issue on Qualitative Vision, Y. Aloimonos (Ed.)*, 14(2):159–178, 1995.

[12] J. Santos-Victor and G. Sandini. Uncalibrated obstacle detection using normal flow. *Machine Vision and Applications*, 9(3):130–137, 1996.

[13] J. Santos-Victor and G. Sandini. Visual behaviors for docking. *Computer Vision and Image Understanding (to appear)*, 1997.

[14] J. Santos-Victor, F. van Trigt, and J. Sentieiro. Medusa - a stereo head for active vision. In *Proc. of the Int. Symposium on Intelligent Robotic Systems*, Grenoble, France, July 1994.

[15] E. Schwartz. Spatial mapping in the primate sensory projection : Analytic structure and relevance to perception. *Biological Cybernetics*, 25:181–194, 1977.

[16] C. Silva and J. Santos-Victor. Direct egomotion estimation. In *Proc. of the 13th Int. Conference on Pattern Recognition*, Vienna,Austria, August 1996.

[17] C. Silva and J. Santos-Victor. Robust egomotion estimation from the normal flow using search subspaces. Technical Report 6/96, ISR/Inst. Sup. Tecnico - VisLab, 1996.

[18] C. Silva and J. Santos-Victor. Robust egomotion estimation from the normal flow using search subspaces. Accepted for Publication by PAMI, 1997.

[19] M.V. Srinivasan, M. Lehrer, W.H. Kirchner, and S.W. Zhang. Range perception through apparent image speed in freely flying honeybees. *Visual Neuroscience*, 6:519–535, 1991.